8-1-2011

Distance metric learning for medical image registration

Zois Boukouvalas

Distance Metric Learning for Medical Image Registration

Zois Boukouvalas

School of Mathematical Sciences

Rochester Institute of Technology

A thesis submitted for the degree of

Master of Science

August 2011

MASTER OF SCIENCE THESIS
FOR
ZOIS BOUKOUVALAS

APPROVED:

Thesis Committee:

Major Professor Dr. Nathan D. Cahill
Dr. James Marengo
Dr. Elizabeth Cherry
Dr. Marvin Gruber

ROCHESTER INSTITUTE OF TECHNOLOGY
2011

Acknowledgements

First, I would like to thank my supervisor Dr. Nathan D. Cahill (Rochester Institute of Technology). His expertise, understanding, and patience, added considerably to my graduate experience. Moreover his knowledge and skill in many areas helped me to understand deeper mathematical concepts associated with their applications in real world problems. I would also like to express my thanks to my committee: Dr. James Marengo, Dr. Elizabeth Cherry and Dr. Marvin Gruber for their assistance and support to my thesis.

For his close friendship and for a number of helpful discussions I have had that have helped me to work on a new for me research area, I would like to thank Dr. Grigorios Tsagkatakis. Furthermore, I would like to thank Professor Patricia Diute for her assistance and the excellent collaboration that I really enjoyed working with her during the spring quarter. Moreover, I would like to thank the dean of our college Dr. Sophia Maggelakis and my previous supervisor Dr. Andreas Arvanitoyeorgos for their encouragement over the years.

For their support and encouragement through my entire life and in particular I would like to thank my parents, Panagiotis and Niki and I would also like to thank my friends. Finally, and most importantly, for her many years of love and beautiful moments we have lived together I would like to dedicate my thesis to Maria Barouti.

Abstract

Medical image registration has received considerable attention in medical imaging and computer vision, because of the large variety of ways in which it can impact patient care. Over the years, many algorithms have been proposed for medical image registration. Medical image registration uses techniques to create images of parts of the human body for clinical purposes. This thesis focuses on one small subset of registration algorithms: using machine learning techniques to train the similarity measure for use in medical image registration. This thesis is organized in the following manner.

In Chapter 1 we introduce the idea of image registration, describe some some applications in medical imaging, and mathematically formulate the three main components of any registration problem: geometric transformation, similarity measure and optimization procedure. Finally we describe how the ideas in this thesis fit into the field of medical image registration, and we describe some related work.

In Chapter 2 we introduce the concept of machine learning and we provide examples to illustrate machine learning algorithms. We then describe the κ-nearest neighbors algorithm and the relationship between Euclidean and Mahalanobis distance. Next we introduce distance metric learning and present two approaches for learning the Mahalanobis distance. Finally we provide a description and visual comparison of two algorithms for distance metric learning.

In Chapter 3 we describe how distance metric learning can be applied to the problem of medical image registration. Our goal is to learn the optimal similarity measure given a training dataset of correctly registered images. To assess the performance of the two distance metric learning algorithms we test them using images from a series of patients. Moreover we illustrate the sensitivity of one of the learning algorithms by examining the variability of the resulting target registration errors. Finally we present our experimental results of registering CT and MR images.

Finally in Chapter 4 we suggest some ideas for future work in order to improve our registration results and to speed up the algorithms.

Contents

List of Figures

Chapter 1

Introduction

Image registration is the process of overlaying two or more images taken at different times or from different viewpoints. It is based on aligning a target image to a source image by determing the transformation that maps points in the target image to points in the source image.

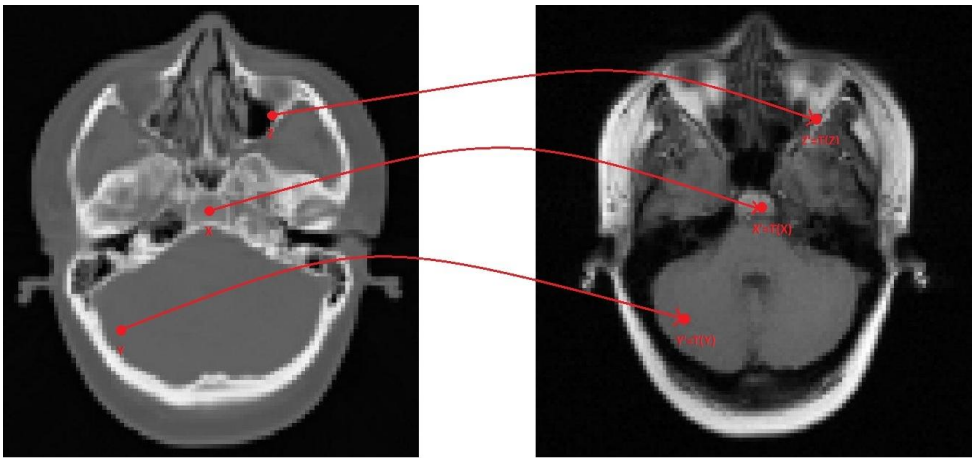


Figure 1.1: Left:target image;right:source image

In medical imaging, the images can come from different modalities including computed tomography (CT), magnetic resonance (MR), ultrasound (US), positron emission tomography (PET), etc. Registration can be performed to align images of either the same or different patients captured from one or more of the modalities. Mathematically, we denote

the target image by $I : \Omega_I \subset \mathbb{R}^2 \to \mathbb{R}$ and the source image by $J : \Omega_J \subset \mathbb{R}^2 \to \mathbb{R}$. Three main components of image registration are the geometric transformation, the similarity measure and the optimization process. The geometric transform maps each point of the target image to the source image. The general form of a transformation can be written as:

$$T : \Omega_I \subset \mathbb{R}^2 \to \Omega_J \subset \mathbb{R}^2, \tag{1.1}$$

where T maps each point $x_k \in \Omega_I$ to $T(x_k) \in \Omega_J$. The similarity measure can be interpreted as a function that quantifies how well the target and the source images are aligned. Finally the optimization procedure defines how to determine the specific transformation of the source image that optimizes the similarity measure between the target image and the transformed source image. Further information about types of transformations, types of similarity measures and the optimization process are given in the next sections.

1.1 Why registration?

There are a wide variety of practical applications of medical image registration. Some specific examples include:

1. **Analysis of temporal evolution**: finding changes or growths in images taken at different times or under different conditions. Registering images of the same patient before and after chemotherapy in order to detect growth of cancer.

2. **Fusion of multimodal images**: integrating information taken from different sensors. Integrating structural information from CT (computed tomography) or MRI (magnetic resonance imaging) with functional information from radionucleic scanners such as PET (positron emission tomography) or SPECT (single-photon emission computed tomography) can facilitate anatomically locating metabolic function.

3. **Inter-patients comparison**: Determining whether a particular anatomical region is normal compared to a given population, for example, to determine if structures in a patient brain are similar to other patients diagnosed with Alzheimer's disease.

1.2 Major components of image registration

There are three major components to any image registration problem: The geometric transformation, the similarity measure and the optimization process.

1.2.1 What type of transformation?

There are different kinds of transformations that we can use to register images depending on the particular application at hand. For the purpose of this thesis, we focus on the rigid transformation, because it is the most basic and is commonly performed to provide a coarse initial registration even when highly deformable transformations are required. Some commonly used transformations are:

1. **Rigid**: Rigid transformations are comprised of only translations and rotations. A rigid transformation can be described using a matrix equation $\mathbf{T}(\mathbf{x}) = \mathbf{R} \cdot \mathbf{x} + \mathbf{t}$, where $\mathbf{R}$ is an $n \times n$ matrix where n is the dimension of the image that denotes the rotation and $\mathbf{t}$ is an $n \times 1$ vector that denotes the translation. Typically, medical images are inherently 3-dimensional quantities, so we will assume that $n = 3$ for the remainder of this thesis. If $\mathbf{x}$ denotes a point in the space of the template image then $\mathbf{T}(\mathbf{x})$ can be represented in vector form by:

$$\begin{pmatrix} T(x_1) \\ T(x_2) \\ T(x_3) \end{pmatrix} = \mathbf{R} \cdot \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} + \begin{pmatrix} t_1 \\ t_2 \\ t_3 \end{pmatrix}, \tag{1.2}$$

where the matrix $\mathbf{R}$ is given by $\mathbf{R} = \mathbf{R}^1\mathbf{R}^2\mathbf{R}^3$, where

$$\mathbf{R}^1 = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta_1) & -\sin(\theta_1) \\ 0 & \sin(\theta_1) & \cos(\theta_1) \end{pmatrix}$$

$$\mathbf{R}^2 = \begin{pmatrix} \cos(\theta_2) & 0 & \sin(\theta_2) \\ 0 & 1 & 0 \\ -\sin(\theta_2) & 0 & \cos(\theta_2) \end{pmatrix}$$

$$\mathbf{R}^3 = \begin{pmatrix} \cos(\theta_3) & -\sin(\theta_3) & 0 \\ \sin(\theta_3) & \cos(\theta_3) & 0 \\ 0 & 0 & 1 \end{pmatrix}.$$

Matrix $\mathbf{R}^i$ rotates the point $\mathbf{x}$ around the i-axis by angle θ_i. Hence, rigid transformations can be expressed with six parameters, three for rotation and three for translation. The set of all rotation matrices forms a group, known as the rotation group or the special orthogonal group denoted SO(3). This group is a subgroup of the orthogonal group O(3), which includes both rotations and reflections and consists of all orthogonal matrices with determinant 1 or -1.

2. **Affine**: Affine transformations map parallel lines onto parallel lines and in 2-D maps triangles to triangles. Affine transformations are given by:

$$\mathbf{T}(\mathbf{x}) = \mathbf{B} \cdot \mathbf{x} + \mathbf{t}, \tag{1.3}$$

where $\mathbf{B}$ is a 3×3 matrix. In three dimensions the affine transformation has twelve

parameters, nine for **B** and three for **t**. Matrix **B** is an element of the general linear group GL(3), which consists of all the three by three matrices with determinant different to zero.

3. **Deformable**: Deformable transformations allow locally changing deformation. **T(x)** can take any general form. A visualization of a deformable transformation is given by the following graph.

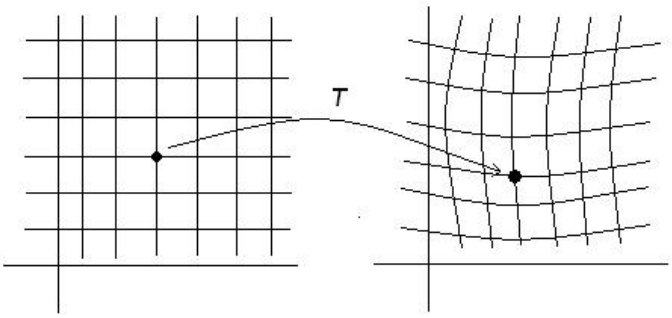


Figure 1.2: Deformable transformation

Examples of deformable transformations are Thin-plate splines, B-spline FFQ, non-parametric vector fields, etc.

Rigid and affine transformations are global in the sense that if a change in any of parameters impacts the transformation everywhere in the image domain, whereas deformable transformations can either be global (thin-plate splines) or local (B-spline FFD, variational). For more details about types of transformations, the reader can consult [9].

1.2.2 What type of similarity?

There are many ways to define the similarity measure, depending on the assumptions underlying the relationship between the source and target images [1]. If I is the target image and J is the source image the following cases can be considered.

1. **Constant relationship**: if the intensity of the two images that we try to register is the same when they are perfectly aligned, then we can use sum/integral of squared difference

$$S(I, J; T) = \int_{\Omega} (I(x) - J(T(x)))^2 \, dx \tag{1.4}$$

or sum/integral of absolute difference

$$S(I, J; T) = \int_{\Omega} |I(x) - J(T(x))| \, dx \tag{1.5}$$

2. **Linear relationship**: if there is a linear relationship between the intensities of the two images when they are perfectly aligned, then we can use normalized cross-correlation as a similarity measure.

$$S(I, J; T)^2 = \frac{\int_{\Omega} (I(x_k) - \bar{I})(J(T(x_k)) - \bar{J}) \, dx}{\sigma_I \sigma_J} \tag{1.6}$$

where $\bar{I}$ and $\bar{J}$ the averages of I and J, respectively given by:

$$\bar{I} = \frac{\int_{\Omega} (I(x_k)) \, dx}{|\Omega|}, \tag{1.7}$$

and

$$\bar{J} = \frac{\int_{\Omega} (J(T(x_k))) \, dx}{|\Omega|}, \tag{1.8}$$

and σ_I, σ_J their standard deviations given by:

$$\sigma_I = \frac{\sqrt{\int_{\Omega} ((I(x_k) - \bar{I})^2 \, dx}}{|\Omega|^{\frac{1}{2}}}, \tag{1.9}$$

and

$$\sigma_J = \frac{\sqrt{\int_\Omega ((J(T(x_k)) - \bar{J})^2 \, dx}}{|\Omega|^{\frac{1}{2}}}. \tag{1.10}$$

3. **Functional relationship**: if there exists an unknown functional relationship between the images when they are perfectly aligned, we can use the correlation ratio, defined by:

$$\eta(J|I) = \frac{Var[E(J|I)]}{Var(J)}, \tag{1.11}$$

where $Var[E(J|I)]$ is the variance of the conditional expectation $E(J|I)$.

4. **Probabilistic relationship**: If there is no functional relationship between image intensities when the images are perfectly aligned, but there may be a probabilistic relationship, a similarity measure can be defined to measure the joint complexity of the images. One such measure is mutual information [6], which can be defined as:

$$MI(I, J) = H(J) - H(J|I), \tag{1.12}$$

where $H(J)$ is the entropy of image J and $H(J|I)$ is the conditional entropy of J given I. Mutual information is most useful in registering images from different modalities such as CT and MR.

1.2.3 Optimization Procedure

The optimization procedure is used to find the transformation parameters that optimize the similarity measure between the target image and the transformed source image.

The following example shows how an optimization procedure based on Newton-Raphson iteration method can be used. A point $\mathbf{x}$ inside the image J will be relocated to $\hat{\mathbf{x}}$ according to the rigid transformation described in 1.2.1:

$$\hat{\mathbf{x}} = \mathbf{T}(\mathbf{x}) = \mathbf{R}\mathbf{x} + \mathbf{t}. \tag{1.13}$$

Recall that the rigid transformation T is composed of six parameters $t_1, t_2, t_3, \theta_1, \theta_2, \theta_3$. The objective is to determine the parameters of T that best align I and J. To optimize S with respect to T, we can identify a critical point by finding the gradient of S with respect to the parameters of T and set it equal to zero; i.e.,

$$\nabla S = 0. \tag{1.14}$$

Equation (1.14) can be solved by using Newton-Raphson iteration method.

$$T^{k+1} = T^k - \nabla S(T^k) \cdot [\nabla\nabla S(T^k)]^{-1}. \tag{1.15}$$

Where $\nabla\nabla S(T^k)$ is the Hessian matrix of the similarity measure evaluated at the current iterate. If the similarity measure is not convex, the minimization can proceed via Levenberg-Marquardt or Quasi-Newton optimization methods. As with any optimization procedure we need to choose an appropriate initial guess. In the absence of any prior information, we choose T^0 to be the identity matrix.

1.3 Our Approach

All of the similarity measures in Sec. 1.2.2 are general in the sense that they can be used in different types of problems as long as their underlying assumptions are valid. However, this general applicability may come at the cost of increased registration accuracy when problem-specific information is available. For instance, if a variety of training examples of correctly-registered images are available in a particular setting, registration could potentially be more accurate if a similarity measure is learned from the training data. One

example of such an approach is [8] where they trained the similarity measure based on the maximum margin structured output learning method using extracted features of the neighborhoods of the target and source images. In this thesis, we will investigate an alternative learning approach namely distance metric learning (DML). The objective of DML is to learn a distance metric that will separate the training data while satisfying a set of distance constraints between the data. Two approaches to perform DML are information theoretic metric learning (ITML) [2] and information geometry (IGML) [14]. Both approaches have the goal of minimizing the distance between two Gaussian distributions which correspond to data from "good" and "bad" training examples. We will apply these algorithms to the problem of CT/MR brain image registration and assess and compare their performance on a publicly available database of CT and MR brain images of a series of patients.

Chapter 2

Machine Learning

The field of machine learning provides methods for *training* a machine (computer) how to *learn* a task using example data or past experience. There are many reasons for training a machine. First, there are situations where no human expertise exists. For instance, a machine learning system can study recorded data and machine failures in order to learn prediction rules. Second, there are situations where human expertise exists, but where most humans are unable to use and explain his/her expertise. Problems that fall in that category are hand-writing recognition and speech recognition. Humans can provide the machine with various of inputs and matched outputs in order to train the machine. Machine learning is extremely useful in a variety of settings for example there are situations such as the stock market where the phenomena are changing very fast, so machine learning can provide a set of prediction rules and for problems like filtering incoming e-mails in order to decide whether messages should be classified as spam. For detailed description of machine learning we suggest [3],[12].

2.1 Learning Categorization

Learning can be divided into two categories: Empirical and analytical. Empirical learning requires human interaction to provide the machine with necessary information for training. Analytical learning, on the other hand does not require human interaction since it improves performance by analyzing the problem. For the purpose of this thesis we will present and analyze empirical learning tasks. Approaches to empirical learning can be described as supervised or unsupervised. In supervised learning, the training examples must be labeled as belonging to a particular class while in unsupervised learning no labels are provided. In the following two sections, we present a detailed explanation of supervised and unsupervised learning.

2.1.1 Supervised Learning

Suppose that we would like to develop an algorithm that can determine if a person in a photograph is happy or sad. Such an algorithm that is used to determine which of two or more classes a particular example belongs to is called a *classifier*. Supervised learning constructs a classifier for a given set of inputs that have been associated with the correct classes. This set of example inputs is called the *training set*. To evaluate the performance of a classifier, we employ a set of examples called the *test set*. Note that classifiers do not have to generate discrete outputs, but can more generally predict numerical values. For example, a classifier can be designed to determine the age and the weight of a person from his/her photograph.

2.1.2 Unsupervised Learning

In unsupervised learning, there are no available class labels, so we can only observe features of a given set of objects. The goal of unsupervised learning is to describe how the data are organized or clustered in order to find a relationship that connects them. One way to

group similar objects together is to define a similarity measure between any two objects and then to cluster them so that all objects in a group are similar according to the defined measure. For example let's assume that we have a set of unlabeled vectors and we want to group them into k-clusters based on the mean of each cluster. We initialize the mean of each cluster and we assign each vector to the cluster with the closest mean. We update the mean of each cluster according to the vectors that we assigned to each cluster. The iteration process stops when the mean remains the same for each iteration.

2.2 Distance Metric Learning for Nearest Neighbor Classification

The κ-nearest neighbors (κ-NN) algorithm is one of the most fundamental algorithms for pattern classification. In a supervised setting the κ-NN algorithm classifies unlabeled examples based on their similarity with examples in the training set. For a given unlabeled example $\mathbf{x}$ we try to find the κ "closest" labeled examples in the training data set and assign $\mathbf{x}$ to the class that appears most frequently within the κ-subset. The κ-NN algorithm requires an integer κ, a set of labebeled examples (training data) and a metric in order to measure the distance between the training data. Typically this metric is chosen to be the Euclidean distance.

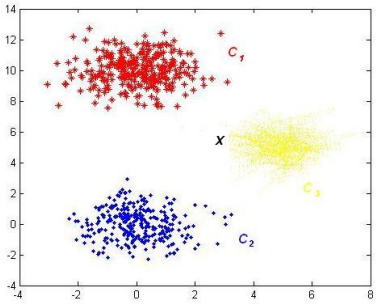


Figure 2.1: Example of (κ-NN) algorithm with three classes, $\kappa = 5$, and the Euclidean distance metric. Of the five closest neighbors to $\mathbf{x}$, three belong to C_3 and two belongs to C_2, so $\mathbf{x}$ is labeled as belonging to C_3.

2.2.1 Mahalanobis Distance

Euclidean distance metrics do not yield good performance for all the types of problems, especially when there is significant correlation in the data. Correlation means that there are associations dependencies in the data. In order to formulate a new distance metric that accounts for correlation in the training data, we first recall the definition of the Euclidean distance. Consider $\mathbf{x} = (x_1, ..., x_p)^T$ and $\mathbf{y} = (y_1, ..., y_p)^T$, both points in $\mathbb{R}^p$. The Euclidean distance between $\mathbf{x}$ and $\mathbf{y}$ is given by

$$d_E(\mathbf{x}, \mathbf{y}) = \sqrt{(\mathbf{x} - \mathbf{y})^T(\mathbf{x} - \mathbf{y})}, \tag{2.1}$$

and the Euclidean norm of $\mathbf{x}$ is given by

$$\|\mathbf{x}\|_2 = d_E(\mathbf{x}, 0) = \sqrt{\mathbf{x}^T\mathbf{x}}. \tag{2.2}$$

It follows from (2.2) that all points with the same distance c from the origin satisfy:

$$x_1^2 + ... + x_p^2 = c^2, \tag{2.3}$$

which is the equation of a $(p-1)$-sphere. This implies that all the components of the point $\mathbf{x}$ contribute equally in determining its Euclidean norm.

Now, consider the case where the points are generated from a multivariate Gaussian distribution in $\mathbb{R}^p$. The multivariate Gaussian distribution is a generalization of the one-dimensional normal distribution to higher dimensions. It can be used to model data which are linearly correlated. If we assume the covariance matrix is a multiple of the identity matrix, the Euclidean distance is adequate to describe distances between data. However, if any components have different variances, or if there is any nontrivial covariance, Eu-

clidean distance fails to account for this type of variability. Thus, in order to measure distances between data generated from a multivariate Gaussian distribution, we use a different metric called *Mahalanobis distance*, which we will describe in the remainder of this section.

First, assume that we have uncorrelated data, but that each component of the data has different variance. Let $\mathbf{x} = (x_1, ..., x_p)^T$ and $\mathbf{y} = (y_1, ..., y_p)^T$ be drawn from a multivariate Gaussian distribution with mean vector μ and covariance matrix $\mathbf{D} = \text{diag}(s_1^2, ..., s_p^2)$. We can shift and scale the components of $\mathbf{x}$ and $\mathbf{y}$ to generate two new vectors $\mathbf{u} = (\frac{x_1}{s_1}, ..., \frac{x_p}{s_p})$ and $\mathbf{v} = (\frac{y_1}{s_1}, ..., \frac{y_p}{s_p})$ which can be considered as points drawn from a multivariate Gaussian distribution with $\mu = \mathbf{0}$ and $\mathbf{\Sigma} = \mathbf{I}$. If we define a distance between $\mathbf{x}$ and $\mathbf{y}$ according to

$$d(\mathbf{x}, \mathbf{y}) = d_E(\mathbf{u}, \mathbf{v}) = \sqrt{(\frac{x_1 - y_1}{s_1})^2 + ... + (\frac{x_p - y_p}{s_p})^2} = \sqrt{(x-y)^T \mathbf{D}^{-1} (x-y)}, \quad (2.4)$$

this new distance measures accounts for the different variances of each component. According to this new distance measure, the norm of $\mathbf{x}$ is equal to

$$\|\mathbf{x}\|_2 = d(\mathbf{x}, 0) = \sqrt{\mathbf{x}^T \mathbf{D}^{-1} \mathbf{x}}. \quad (2.5)$$

Hence, we can say that points with same distance of the origin satisfy

$$(\frac{x_1}{s_1})^2 + ... + (\frac{x_p}{s_p})^2 = c^2, \quad (2.6)$$

which is the equation of an ellipsoid centered at the origin.

Now, consider that the data are also linearly correlated and that $\mathbf{A}$ is the covariance

matrix of the multivariate Gaussian random variable. Our goal is to stretch and rotate the space where the variables live, in order to reflect the correlation in the data. Let $\mathbf{x} = (x_1, ..., x_p)^T$ and $\mathbf{y} = (y_1, ..., y_p)^T$ be drawn from a multivariate Gaussian distribution. We shift, scale and rotate $\mathbf{x}$ and $\mathbf{y}$ to generate new vectors $\bar{\mathbf{x}} = \mathbf{L}^{-1}\mathbf{x}$ and $\bar{\mathbf{y}} = \mathbf{L}^{-1}\mathbf{y}$, where $\mathbf{L}$ is the Cholesky factor of $\mathbf{A}$. According to (2.4) the distance between $\bar{\mathbf{x}}$ and $\bar{\mathbf{y}}$ is given by

$$\begin{aligned} d(\bar{\mathbf{x}}, \bar{\mathbf{y}}) &= \sqrt{(\bar{\mathbf{x}} - \bar{\mathbf{y}})^T(\bar{\mathbf{x}} - \bar{\mathbf{y}})} \\ &= \sqrt{[(\mathbf{L}^{-1})(\mathbf{x} - \mathbf{y})]^T[\mathbf{L}^{-1}(\mathbf{x} - \mathbf{y})]} \\ &= \sqrt{((\mathbf{x} - \mathbf{y})^T\mathbf{L}^{-T}\mathbf{L}^{-1}(\mathbf{x} - \mathbf{y}))} \\ &= \sqrt{((\mathbf{x} - \mathbf{y})^T\mathbf{A}^{-1}(\mathbf{x} - \mathbf{y}))}. \end{aligned} \tag{2.7}$$

Since $\mathbf{A} = \mathbf{L}\mathbf{L}^T$ (2.7) simplifies to

$$d_{\mathbf{A}}(\mathbf{x}, \mathbf{y}) = \|\mathbf{L}^{-1}(\mathbf{x} - \mathbf{y})\|_2. \tag{2.8}$$

This is the *Mahalanobis distance*, which is equivalent to the Euclidean distance computed after we change the basis of our space.

In many machine learning problems, it is reasonable to assume that training data has nontrivial correlations. Using Mahalanobis distance as a model, the objective of distance metric learning (DML) is to learn that matrix $\mathbf{A}$ from the training data. DML has received considerable attention in the research literature [15] and a number of different DML approaches have been presented. The goal of (DML) is to train a distance metric in order to seperate the data according to their class labels.

In order to present some of the methods which are used for (DML) we introduce in the next section some concepts from information geometry, including manifolds, tangent

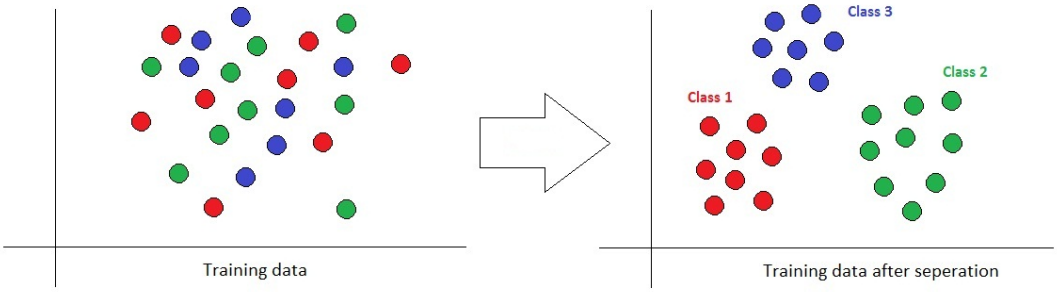


Figure 2.2: The first figure shows the projection of a high dimensional dataset into two dimensions; the figure at the right shows the result of DML performed in the higher dimension and then projected into two dimensions.

space and Riemannian manifolds.

2.2.2 Elements of Information Geometry

Information geometry is the field that connects differential geometry and probability. Let

$$S = \{N(\mu, \sigma^2) | \mu \in \mathbb{R}, \sigma^2 \in \mathbb{R}^+\}, \tag{2.9}$$

be the set of Gaussian distribution with mean μ and variance σ^2, with probability densities given by:

$$p(x; \mu, \sigma) = \frac{1}{\sqrt{2\pi}\sigma} \exp\{\frac{(x-\mu)^2}{2\sigma^2}\}. \tag{2.10}$$

Specific values of μ and σ determine a particular Gaussian distribution, so S can be thought of as a two dimensional space having a (μ, σ) coordinate system. S is not, however, a Euclidean space, but a Riemannian space. For a very brief overview, the next subsections introduce the concepts of manifold, tangent space and Riemannian space. For more details we suggest [11],[4].

2.2.2.1 Manifold

A differential manifold is a generalization of a geometric object in a high dimensional space. A coordinate system is a one-to-one mapping from the manifold to $\mathbb{R}^n$. Let S be a manifold having coordinate system $\phi : S \to \mathbb{R}^n$. Then ϕ maps each $p \in S$ to an n-type of real numbers:

$$\phi(p) = [\xi^1(p), ..., \xi^n(p)] = [\xi^1, ..., \xi^n] = \xi. \tag{2.11}$$

ξ contains the coordinates of the point p, and each ξ^i is a *coordinate function* that maps a point p to its i^{th} coordinate. Let $\psi = [\rho^i]$ be another coordinate system for S. A point $p \in S$ has coordinates with respect to both ϕ and ψ. As we did for the ϕ coordinate system, we define a set of coordinate functions with respect to the ψ coordinate system.

To provide a formal definition of a manifold, let $\mathcal{A}$ be the set of all coordinate systems. Now define S equipped with $\mathcal{A}$ to be an n-dimensional manifold if the following conditions are satisfied.

1. Each $\phi \in \mathcal{A}$ is a one-to-one mapping from S to some open subset of $\mathbb{R}^n$.

2. For all $\phi \in \mathcal{A}$, given any one-to-one mapping ψ from S to $\mathbb{R}^n$, the following holds:

$$\psi \in \mathcal{A} \iff \psi \circ \phi^{-1} \text{ is a } C^\infty \text{ diffeomorphism.}$$

The following example gives a nice view of the definition of a differential manifold. Let $S = S^2 = \{(x_1, x_2, x_3) \in \mathbb{R}^3 \mid x_1^2 + x_2^2 + x_3^2 = 1\}$. We prove that $(S, \mathcal{A})$ is a differential manifold. We define two open subsets U and V by excluding the north and the south pole. Thus we have $U = S^2 - (0, 0, 1)$ and $V = S^2 - (0, 0, -1)$. We can obtain two coordinate systems by using stereographical projections from the north and south pole.

$$\phi : U \to \mathbb{R}^2,\ \phi(x_1, x_2, x_3) = \left(\frac{x_1}{1 - x_3}, \frac{x_2}{1 - x_3} \right), \tag{2.12}$$

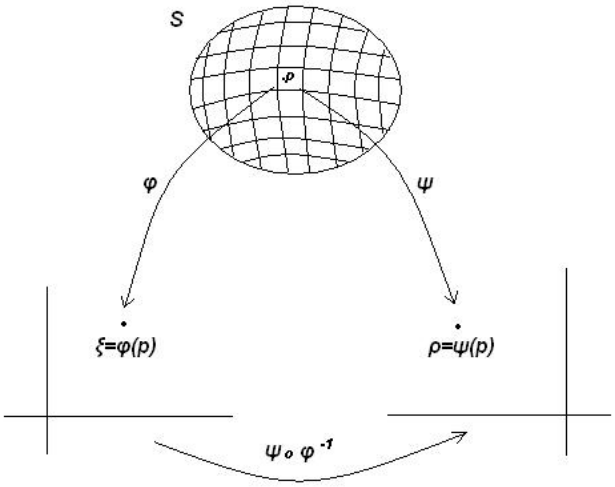


Figure 2.3: Geometrical view of a differential manifold

and

$$\psi : V \to \mathbb{R}^2,\ \psi(x_1, x_2, x_3) = \left(\frac{x_1}{1 + x_3}, \frac{x_2}{1 + x_3} \right). \tag{2.13}$$

We can observe that (2.3), (2.4) are one-to-one from S to some open subset of $\mathbb{R}^2$. Moreover we have to define ϕ^{-1} and ψ^{-1}. Thus we have

$$\phi^{-1}(u_1, u_2) = \left(\frac{2u_1}{u_1^2 + u_2^2 + 1}, \frac{2u_2}{u_1^2 + u_2^2 + 1}, \frac{u_1^2 + u_2^2 - 1}{u_1^2 + u_2^2 + 1} \right), \tag{2.14}$$

and

$$\psi^{-1}(v_1, v_2) = \left(\frac{2v_1}{v_1^2 + v_2^2 + 1}, \frac{2v_2}{v_1^2 + v_2^2 + 1}, \frac{1 - v_1^2 - v_2^2}{v_1^2 + v_2^2 + 1} \right). \tag{2.15}$$

According to the definition we have to show that $\psi \circ \phi^{-1}$ and $\phi \circ \psi^{-1}$ are C^∞ diffeomorphisms. Note that

$$\psi \circ \phi^{-1}(u_1, u_2) = \left(\frac{u_1}{u_1^2 + u_2^2}, \frac{u_2}{u_1^2 + u_2^2} \right), \tag{2.16}$$

and

$$\phi \circ \psi^{-1}(v_1, v_2) = \left(\frac{v_1}{v_1^2 + v_2^2}, \frac{v_2}{v_1^2 + v_2^2} \right), \tag{2.17}$$

which are C^∞ diffeomorphisms, hence S^2 is a differential manifold.

2.2.2.2 Tangent Space

For every point $p \in S$, there is an n-dimensional hyperplane called the tangent space T_pS which is the best linear approximation of S near p. In order to define the T_pS for a particular manifold S, we should first introduce the idea of tangent vector.

Let $\gamma : I \subset \mathbb{R} \to \mathbb{R}^n$ be a smooth curve. The derivative of this curve at a point p is defined to be the derivative $\frac{\mathrm{d}\gamma}{\mathrm{d}t}|_p$. The tangent space at point p is defined as the one-dimensional vector space generated by the tangent vector $\frac{\mathrm{d}\gamma}{\mathrm{d}t}|_p$. Now assume that $g : J \subset \mathbb{R}^2 \to \mathbb{R}^n$ is a smooth surface. The tangent vectors $\frac{\partial g}{\partial u}|_p$, $\frac{\partial g}{\partial v}|_p$ at point p generate a two-dimensional vector space that is the tangent space of g at p. Our goal is to define the tangent vectors of a differential manifold without assuming that the manifold is embedded in a Euclidean space.

Let S be a differential manifold equipped with a set of coordinate systems. A tangent vector of S at p is defined as a smooth real function

$$u : S_p \to \mathbb{R}, \tag{2.18}$$

and satisfies the following two conditions

1.

$$u(\lambda f + \mu g) = \lambda u(f) + \mu u(g) \tag{2.19}$$

2.

$$u(fg) = u(f)g(p) + f(p)u(g) \tag{2.20}$$

where f, g are smooth functions and λ, μ are real numbers. The set of all tangent vectors at p is called *tangent space*.

2.2.2.3 Riemannian Manifolds

Let S be a manifold. For any pount $p \in S$ we define $\langle , \rangle_p$ to be the inner product on the tangent space T_pS. The inner product for any tangent vectors $D, D' \in T_pS$ satisfies the following conditions.

1. **Symmetry**

$$\langle D, D' \rangle_p = \langle D', D \rangle_p \tag{2.21}$$

2. **Positive-definitness**

$$\text{If } D \neq 0 \text{ then } \langle D', D \rangle_p > 0 \tag{2.22}$$

3. **Linearity**

$$\langle aD + bD', D'' \rangle_p = a\langle D, D'' \rangle_p + b\langle D', D'' \rangle_p \text{ , where } a, b \in \mathbb{R} \tag{2.23}$$

The above conditions make the inner product bilinear and the set of all inner products on T_pS is defined as:

$$B(T_pS) = \{f : T_pS \times T_pS \to \mathbb{R} \mid f \text{ bilinear}\}, \tag{2.24}$$

where f maps a set of tangent vectors to a real number via the inner product. The map $g : S \to B(T_pS)$ is called a Riemannian metric and every differential manifold equipped with this metric is called *Riemannian manifold*.

If we define $[\xi^i]$ to be a coordinate system for S and $\partial_i = (\frac{\partial}{\partial \xi^i})_p$ to be a tangent vector, then the components of the Riemannian metric on a point $p \in S$ are given by:

$$g_{ij}(p) = \langle \partial_i, \partial_j \rangle_p, \tag{2.25}$$

Equation (2.21) and (2.22) reveal that the matrix $G = [g_{ij}(p)]$ which is created by the components of the Riemannian metric is a symmetric positive definite matrix.

2.3 Information Theoretic Metric Learning

The goal of Distance Metric Learning (DML) is to learn a distance metric that will separate the training data while satisfying the constraint that data from similar classes will be closer to each other than data from different classes. Since there are typically correlations in data, the Mahalanobis distance is chosen as the metric, and the task of DML becomes learning the matrix $\mathbf{A}$ parameterizing the Mahalanobis distance. Information theoretic metric learning (ITML) [2] is one technique for learning this matrix $\mathbf{A}$.

Suppose the training data is comprised of a set of d points $\{x_1, x_2, ..., x_d\} \in \mathbb{R}^d$ and suppose we have an initial estimate $\mathbf{A}_0$ of the matrix parameterizing the Mahalanobis distance. ITML searches for the matrix $\mathbf{A}$ that separates the data while remaining as close as possible to $\mathbf{A}_0$. In order to find the optimal matrix $\mathbf{A}$ we use the relation between the set of symmetric positive definite matrices and the set of multivariate Gaussian distributions. A given Mahalanobis distance parameterized by $\mathbf{A}$ corresponds to a multivariate Gaussian distribution with probability density function

$$p(\mathbf{x}; \mathbf{A}) = \frac{1}{Z} \exp(-\frac{1}{2} d_{\mathbf{A}}(\mathbf{x}, \mu)), \tag{2.26}$$

where Z is a normalizing constant, μ is the mean and $\mathbf{A}^{-1}$ is the covariance of the Gaussian distribution. The measure that we are using to define the distance between $p(\mathbf{x};\mathbf{A})$ and $p(\mathbf{x};\mathbf{A}_0)$ is the Kullback-Leibler divergence.

$$\mathrm{KL}(p(\mathbf{x};\mathbf{A}_0)||p(\mathbf{x};\mathbf{A})) = \int p(\mathbf{x};\mathbf{A}_0)\log\frac{p(\mathbf{x};\mathbf{A}_0)}{p(\mathbf{x};\mathbf{A})}\,dx. \tag{2.27}$$

Two points from the training data are similar if $d_{\mathbf{A}}(x_i,x_j) \leq u$ for small value of u and disimilar if $d_{\mathbf{A}}(x_i,x_j) \geq l$ for large l. Thus for those two classes of points we formulate the minimization problem as follows:

$$\min_{\mathbf{A}} \mathrm{KL}(p(\mathbf{x};\mathbf{A}_0)||p(\mathbf{x};\mathbf{A}))$$

subject to

$$d_{\mathbf{A}}(x_i,x_j) \leq u,\ x_i \text{ and } x_j \text{ from same class} \tag{2.28}$$

$$d_{\mathbf{A}}(x_i,x_j) \geq l,\ x_i \text{ and } x_j \text{ from different classes} \tag{2.29}$$

Where the constraints are defined regarding to the relationship of the training data.

2.4 Information Geometry for Disatance Metric Learning

An alternative approach for distance metric learning is information geometric learning (IGML)[14]. The idea of IGML is to construct two symmetric positive definite matrices, one based on the class labels on the class labels assigned to the data and the other based on the distances between the training data. The matrix $\mathbf{A}$ that seperates the training data is obtained by minimizing the KL-divergence between the two constructed matrices.

Let $\mathbf{X} = \{x_1, x_2, ..., x_d\}$ be a set of d points in an n-dimensional space, and $\mathbf{Y} = \{y_1, y_2, ..., y_d\}$ is the matrix that contains 0 and 1 which are the class labels assigned to the training data. To determine the appropriate Mahalanobis matrix $\mathbf{A}$, we construct a symmetric positive definite matrix $\mathbf{K}_X$ which is a function of the distance matrix $\mathbf{A}$ and the training data $\mathbf{X}$, and then we attempt to solve for $\mathbf{A}$, and a symmetric positive definite matrix $\mathbf{K}_D$ based on the class labels.

Construction of $\mathbf{K}_X$

For a given training set $\mathbf{X}$ we define $\mathbf{A}$ to be the Mahalanobis distance matrix. First define $\mathbf{M} = \mathbf{A}^{\frac{1}{2}}$ and using the fact that there exists a linear tranformation that maps x to $\mathbf{M}x$, we define

$$\mathbf{K}_X = (\mathbf{MX})^T(\mathbf{MX}) = \mathbf{X}^T\mathbf{M}^T\mathbf{MX} = \mathbf{X}^T\mathbf{AX}. \tag{2.30}$$

Construction of $\mathbf{K}_D$

By using the set of class labels we construct the matrix $\mathbf{YY}^T$. Since $\mathbf{YY}^T$ is a singular matrix, we construct the nearby matrix:

$$\mathbf{K}_D = \mathbf{YY}^T + \lambda\mathbf{I}_n, \tag{2.31}$$

where $\lambda > 0$ is a constant that guarantees that $\mathbf{K}_D$ is nonsingular.

The distance between two positive definite matrices $\mathbf{P}$ and $\mathbf{Q}$ is equal to

$$d(\mathbf{P}||\mathbf{Q}) = \frac{1}{2}(\mathrm{tr}(\mathbf{Q}^{-1}\mathbf{P}) + \log|\mathbf{Q}| - \log|\mathbf{P}| - n). \tag{2.32}$$

Thus we have

$$d(\mathbf{K}_X||\mathbf{K}_D) = \frac{1}{2}(\mathrm{tr}(\mathbf{K}_D^{-1}\mathbf{K}_X) + \log|\mathbf{K}_D| - \log|\mathbf{K}_X| - n). \tag{2.33}$$

The optimal solution of $\mathbf{A}$ is given by the following minimization problem.

$$\min_{\mathbf{A}} d(\mathbf{K}_X||\mathbf{K}_D). \tag{2.34}$$

Using (2.33) we have

$$d(\mathbf{K}_X||\mathbf{K}_D) = \min_{\mathbf{A}} \operatorname{tr}(\mathbf{K}_D^{-1}\mathbf{X}^T\mathbf{A}\mathbf{X}) - \log|\mathbf{A}|, \tag{2.35}$$

and the optimal solution of the above minimization problem is the following

$$\mathbf{A} = (\mathbf{X}\mathbf{K}_D^{-1}\mathbf{X}^T)^{-1}. \tag{2.36}$$

For real world problems the above solution may be computationaly expensive. For that reason we use *Sherman-Morrison* lemma in order to make this expression simpler.

$$\begin{aligned} \mathbf{K}_D^{-1} &= (\mathbf{Y}^T\mathbf{Y})^{-1} - (\mathbf{Y}^T\mathbf{Y})^{-1}\lambda\left(\mathbf{I}_n + \mathbf{I}_n(\mathbf{Y}^T\mathbf{Y})^{-1}\lambda\right)^{-1}\mathbf{I}_n(\mathbf{Y}^T\mathbf{Y})^{-1} \\ &= (\mathbf{Y}^T\mathbf{Y})^{-1} - (\mathbf{Y}^T\mathbf{Y})^{-1}\lambda\left(\mathbf{I}_n + (\mathbf{Y}^T\mathbf{Y})^{-1}\lambda\right)^{-1}(\mathbf{Y}^T\mathbf{Y})^{-1}. \end{aligned} \tag{2.37}$$

Applying *Sherman-Morrison* lemma to $\left((\mathbf{Y}^T\mathbf{Y})^{-1}\lambda + I_n\right)^{-1}$ we obtain the following:

$$\begin{aligned} ((\mathbf{Y}^T\mathbf{Y})^{-1}\lambda + \mathbf{I}_n)^{-1} &= ((\mathbf{Y}^T\mathbf{Y})^{-1}\lambda)^{-1} - ((\mathbf{Y}^T\mathbf{Y})^{-1}\lambda)^{-1}\left(\mathbf{I}_n + ((\mathbf{Y}^T\mathbf{Y})^{-1}\lambda)^{-1}\right)^{-1}\left((\mathbf{Y}^T\mathbf{Y})\lambda\right)^{-1} \\ &= \lambda^{-1}(\mathbf{Y}^T\mathbf{Y}) - \lambda^{-1}(\mathbf{Y}^T\mathbf{Y})\left(\mathbf{I}_n + (\mathbf{Y}^T\mathbf{Y})\lambda^{-1}\right)^{-1}\ \lambda^{-1}(\mathbf{Y}^T\mathbf{Y}). \end{aligned} \tag{2.38}$$

Using (2.37) and (2.38) we have the final result

$$\mathbf{K}_D = \left(\mathbf{I}_n + (\mathbf{Y}^T\mathbf{Y})\lambda^{-1}\right)^{-1} \cdot \lambda^{-1}. \tag{2.39}$$

Thus the optimal solution can be written as:

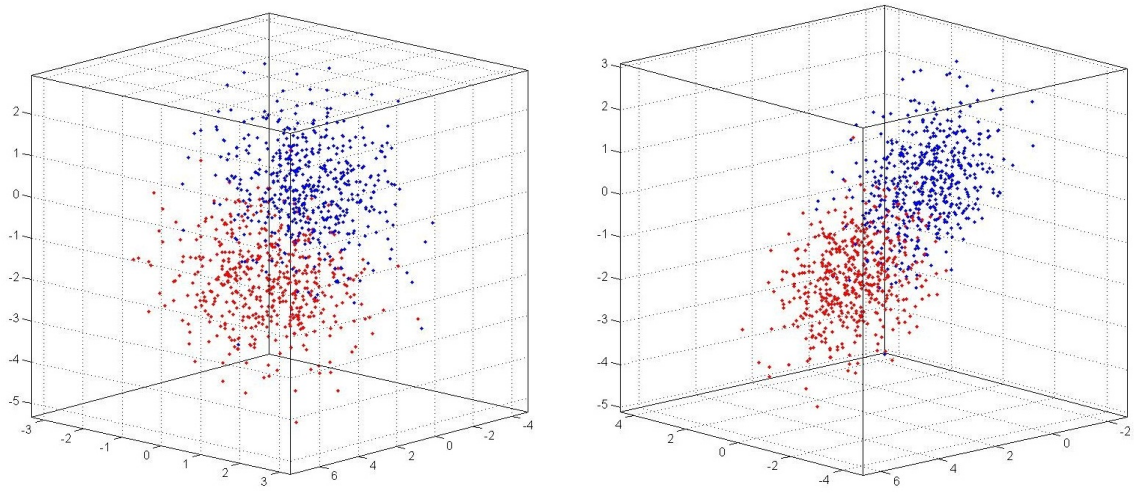
$$\mathbf{A} = (\mathbf{X}\lambda(\mathbf{I}_n + (\mathbf{Y}^T\mathbf{Y})\lambda^{-1})\mathbf{X}^T)^{-1}. \tag{2.40}$$

2.5 ITML and IGML for low dimensional training data

To illustrate the behavior of the ITML and the IGML algorithms, we performed an experiment in three dimensions. For the purpose of our experiment we collected randomly generated data associated with their classes. The goal was to construct a matrix $\mathbf{A}$ using ITML and IGML algorithms to in order to seperate them based on their class labels.

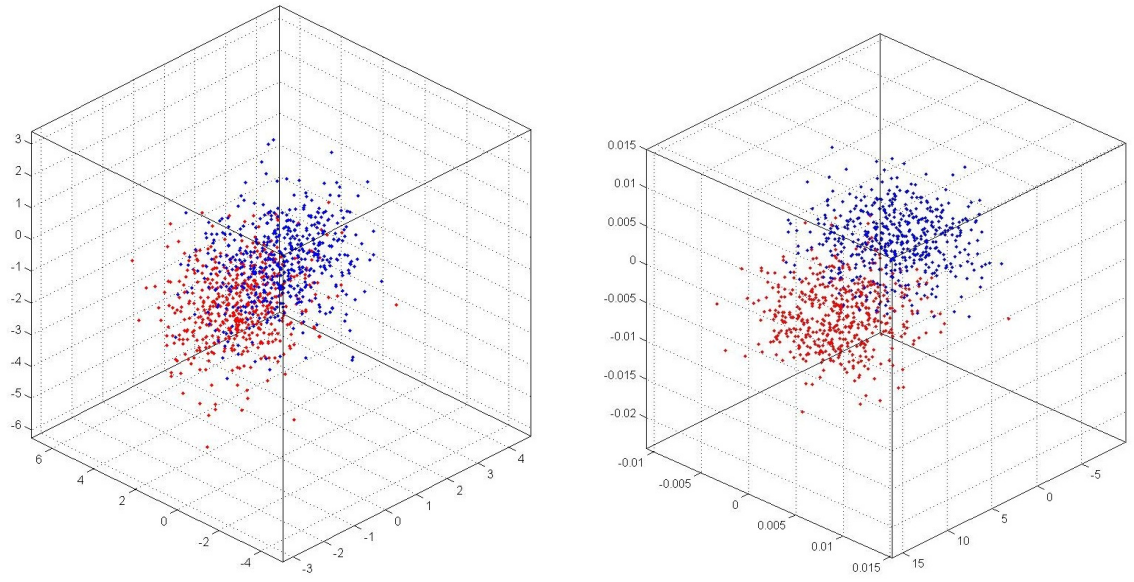
2.5.1 Experimental Results

The following two graphs show the behavior of the randomly generated three-dimensional data before and after we apply ITML and IGML algorithms. The left figure shows two classes of training data that are randomly distributed in the space. The right figure shows results of transforming the training data by the Cholesky factor of the learned matrix $\mathbf{A}$.



(a) Training data before ITML (b) Training data after ITML

Figure 2.4: Class 1 (red) and class 2 (blue) points before (left) and after (right) ITML.



(a) Training data before IGML (b) Training data after IGML

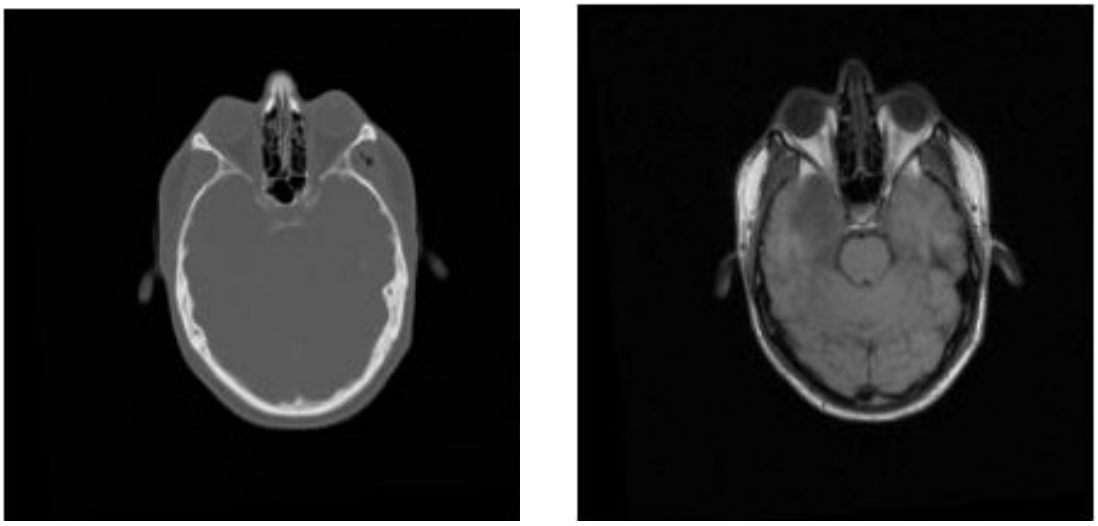
Figure 2.5: Class 1 (red) and class 2 (blue) points before (left) and after (right) IGML.

Chapter 3

Application to Medical Image Registration

In this chapter we apply ITML and IGML to the problem of medical image registration. As we mentioned in the first chapter, a fundamental step in image registration is the selection of the similarity measure. Instead of using a general similarity measure, we learn a similarity measure from our training data. To assess the performance of ITML and IGML we use images from the Retrospective Image Registration Evaluation project [10] which contains CT, MR and PET brain images for a series of different patients.

For this thesis, we use the CT image as the source image and MR image as the target image. For the training algorithm we use CT and MR images from one of the patients for which the RIRE project has provided known ground truth registration results. For the testing algorithm we use the images of six other patients.



(a) CT (b) MR

Figure 3.1: Axial slices of CT and MR images from patient five.

3.1 Learn Similarity Measure

Let $I : \Omega_I \subset \mathbb{R}^2 \to \mathbb{R}$ be the target (CT) image, and $J : \Omega_J \subset \mathbb{R}^2 \to \mathbb{R}$ be the source (MR) image. Our goal is to find a transformation $T : \Omega_I \subset \mathbb{R}^2 \to \Omega_J$ that optimizes a similarity measure over a valid set of transformations. The optimum transformation can be defined by solving the following minimization problem:

$$\min_{T \in \mathcal{T}} S(T). \tag{3.1}$$

For training our similarity measure, we need to construct a training set G of correctly registered *patches* (local pixel neighborhood in each image). One can extract patches from the entire image, but since interesting information is located in only some regions of the image, we focus only on regions of the image that contains information with high contrast. The selection of these regions is based on the gradient of the image.

$$\Omega = \{p \in \Omega_I, \Omega_J \mid ||\nabla I(p)||, ||\nabla J(T(p))|| > \theta\}, \tag{3.2}$$

where $\nabla I(p)$ and $\nabla J(T(p))$ denote the target and source image gradient at a point p and θ is a threshold parameter.

3.1.1 Learning Similarity using ITML and IGML

In our method we use ITML and IGML algorithms in order to train the similarity function for multi-modal medical image registration. Our algorithm proceeds in the following manner:

Step 1

Isotropically sample the target and source image from training set.

Step 2

Construct the set Ω from (3.2).Since we do not need the entire domain of the image we restrict the region according to (3.2). Define the $N \times N \times N$ where the magnitude of the gradient is larger than the threshold.

Step 3

Construct a list of "good" patches by unwrapping and appending correctly registered patches in CT and MR images. Construct a list of "bad" patches by unwrapping and appending incorrectly registered patches (these can be randomly selected). Good and bad patches should be represented as $2N^3 \times 1$ vectors.

Step 4

Use ITML or IGML in order to compute the Mahalanobis matrix as described in chapter 2.

The following diagram is the flow chart of the training algorithm.

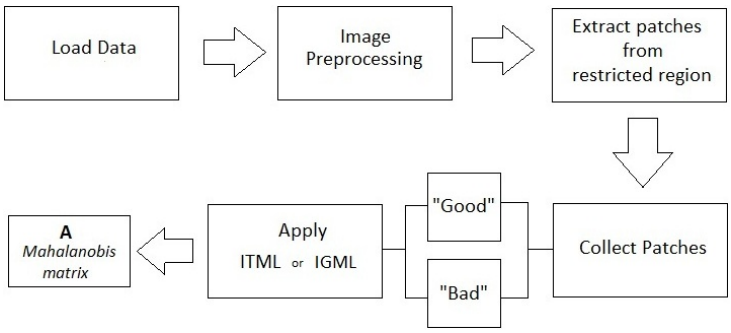
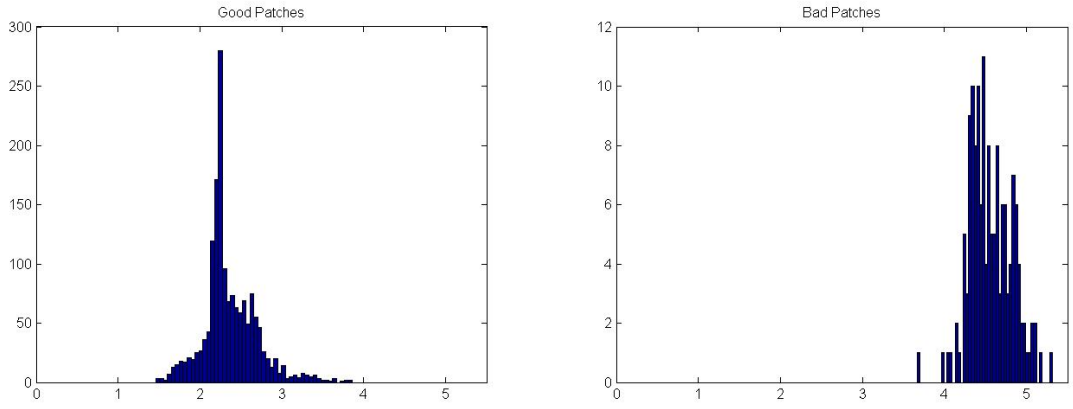


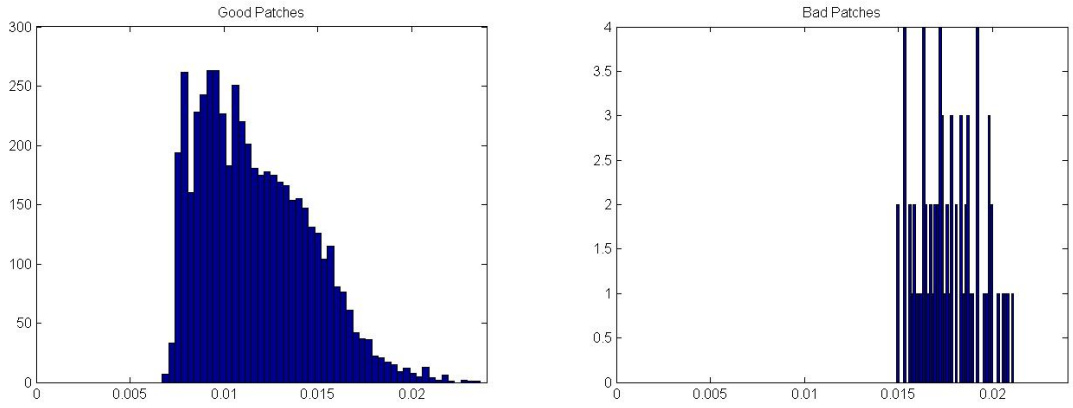
Figure 3.2: Flow chart of training algorithm

Once we determine the Mahalanobis matrix **A**, we can visualize the separation between two classes in the training data by constructing histograms of interpoint Mahalanobis distances. Figure 3.3 shows the frequency between the norms of each good and bad patch, using ITML. Figure 3.4 shows the frequency between the norms of each good and bad patch, using IGML.



(a) Good Patches (b) Bad Patches

Figure 3.3: Separation of good and bad patches using ITML



(a) Good Patches (b) Bad Patches

Figure 3.4: Separation of good and bad patches using IGML

3.1.2 Variability in Results from ITML

ITML is a stochastic algorithm; it uses a random selection of the training examples to compute the Mahalanobis matrix **A**. Therefore, there may be some variability in the results of ITML which we have examined in the following way:

Step 1

Perform ITML twenty times to generate various possible Mahalanobis matrices.

Step 2

Use each possible Mahalanobis matrix to construct the learned similarity measure, and then perform image registration with the resulting similarity measure.

Every time registration is performed, the resulting target registration error (TRE) may be different. A good measure of this sensitivity of ITML for registration is the variability in TRE. Each box contains the 50th percentile of the TRE and the upper edge and lower edge indicate the 75th and 25th percentile of the TRE respectively.The red line inside the box indicates the median value of the TRE. The ends of the vertical lines indicate the minimum and maximum TRE values. We observe that for patient one and six the variability of the errors is around the median value. However, we observe significant

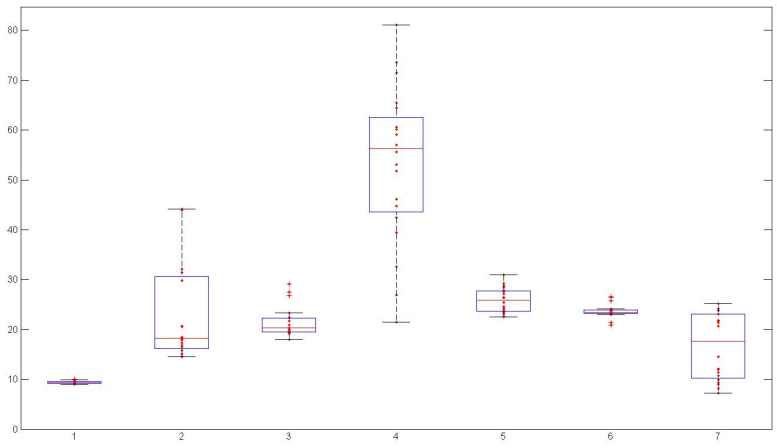


Figure 3.5: Box plot of the different similarity measures for each patient

variability in TRE for the remaining patients. This suggests that ITML is not robust for the task of image registration.

3.2 Experimental results

We applied ITML and IGML algorithms to learn an optimal similarity measure for registering CT and MR images from the RIRE dataset. The threshold parameter θ was selected to 0.2, and N was chosen to be three, yielding $3 \times 3 \times 3$ patches. Target registration error (TRE) was computed after performing registration for each patient. TRE values are listed in the following tables.

CT-T1	**VOL 1**	**VOL 2**	**VOL 3**	**VOL 4**	**VOL 5**	**VOL 6**	**VOL 7**	**VOL 8**	**VOL 9**	**VOL 10**
Patient 1	12.5219	12.1255	12.3049	7.56847	10.6246	6.82679	12.7237	11.4325	11.5664	14.2706
Patient 2	-	24.1708	34.8721	28.1793	24.8749	21.874	32.8282	25.8744	23.911	28.0355
Patient 3	-	24.0293	13.2492	13.0615	16.7834	8.17091	8.41181	28.9039	33.2249	33.3963
Patient 5	20.5433	20.8751	39.7357	32.8517	23.2356	24.6185	36.3093	25.9675	27.4159	20.8001
Patient 6	26.1429	26.198	27.6498	28.0279	26.7834	27.4441	27.0351	26.505	25.6988	25.5245
Patient 7	-	5.15924	14.7344	15.3932	5.73144	12.924	10.6045	9.05258	9.83163	8.39097

Table 3.1: Registration results using ITML

CT-T1	VOL 1	VOL 2	VOL 3	VOL 4	VOL 5	VOL 6	VOL 7	VOL 8	VOL 9	VOL 10
Patient 1	8.05492	8.19051	8.59419	8.62411	8.28666	8.61267	8.60854	9.32382	8.39666	8.42605
Patient 2	-	9.91426	11.3648	11.4737	10.3925	10.894	10.7944	10.1572	9.41594	9.40933
Patient 3	-	4.18751	5.03131	5.13794	4.47013	4.93824	4.84499	4.64035	3.95549	3.95299
Patient 5	11.8326	12.0084	13.0727	12.6247	12.1652	12.4861	13.1274	13.3842	12.3542	11.9674
Patient 6	18.1605	18.2201	19.5653	19.9617	18.7637	19.4338	18.9996	18.6963	17.7976	17.6142
Patient 7	-	2.27654	2.48116	2.35653	2.27281	2.4006	2.67271	2.96102	2.31796	2.49253

Table 3.2: Registration results using IGML

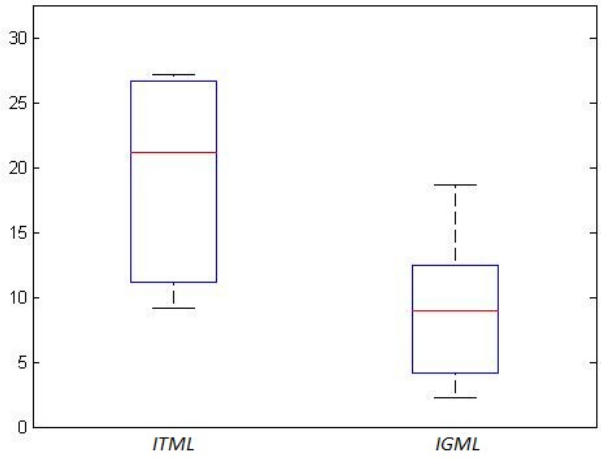


Figure 3.6: Box plot of registration results using IGML and ITML

Comparing those two methods for image registration shows that learning a similarity measure with IGML yields better target registration errors than ITML.

Chapter 4

Conclusion

As shown in chapter 3, IGML outperforms ITML for the task of CT and MR image registration. However there is still room for improvement. To improve the quality of the results we suggest the following methods which are divided into two components. On the first component we will discuss about different approaches for the learned similarity measure and on the second component we will talk about methods that can speed up the learning algorithms.

4.1 Kernel based methods

In chapter 2 we described how to train a linear distance metric based on two kernel matrices constructed by the class labels and a distance metric matrix. In this section, we describe some suggestions of ideas drawn from the field of machine learning.

4.1.1 Learning Metric with nonlinear kernels

Learning metric with nonlinear kernels [14] is an extension of linear metric learning introducing a kernel function. The framework is described by the following steps:

Step 1

Let $\mathbf{X}_1 = (x_1, x_2, ..., x_n)$ denote the collection of n training examples. We define a nonlinear kernel which is actually a function of the form

$$\kappa(x, x') : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}, \tag{4.1}$$

and can be interpreted as the similarity of x, x'.

Step 2

We define a mapping ϕ that maps each point from the input space to an inner product space named feature space $\mathcal{H}_\kappa$.

$$\phi : \mathbb{R}^n \to \mathcal{H}_\kappa. \tag{4.2}$$

Using the above equation we can rewrite the training examples in the following form:

$$\mathbf{X}_1 = (\phi(x_1), \phi(x_2), ..., \phi(x_n)) \,. \tag{4.3}$$

Step 3

We construct the kernel matrix $\mathbf{K}$ by using the inner product

$$\mathbf{K}_{ij} = \kappa(x_i, x_j) = \langle \phi(x_i), \phi(x_j) \rangle, \;\; i, j = 1, ..., n. \tag{4.4}$$

Step 4

We generate a new data representation using the linear operator $M : \mathcal{H}_\kappa \to \mathbb{R}^n$. Thus the data can be written in the new form of

$$\mathbf{X}_2 = M\mathbf{X}_1 = (M\phi(x_1), M\phi(x_2), ..., M\phi(x_n)) \,. \tag{4.5}$$

Following the above steps we constructed $\mathbf{K}_X$ as:

$$\mathbf{K}_X = \mathbf{X}_2^T \mathbf{X}_2 = \mathbf{X}_1^T M^T M \mathbf{X}_1 = \mathbf{X}_1^T \mathbf{A} \mathbf{X}_1. \tag{4.6}$$

By using the result from (2.37) matrix $\mathbf{A}$ is given by:

$$\mathbf{A} = \left(\mathbf{X}_1 \mathbf{K}_D^{-1} \mathbf{X}_1^T\right)^{-1}. \tag{4.7}$$

In the original paper described IGML [14] they demonstrated that kernel-based IGML was superior to linear IGML for some applications. Hence it would be useful to investigate whether or not kernel-based IGML could further improve the results in medical image registration.

4.2 Random Projections

All of these learning techniques are computationally expensive. We could potentially speed up the algorithms using random projections [5],[13]. Random projections are used for dimensionality reduction in a manner that doesn't affect the structure of the input space.

Let $\mathbf{D} \in \mathbb{R}^{m \times n}$ be the matrix that describes the data. The objective is to define a matrix $\mathbf{R}$ of dimension $k \times n$ ($k << m$) such that

$$\mathbf{B} = \frac{1}{\sqrt{n}} \mathbf{R}\mathbf{A} \in \mathbb{R}^{k \times n}. \tag{4.8}$$

The Johnson-Lindenstrauss (JL) lemma [5] shows that the distances of the data in the high dimensional space will preserve invariant under the action of the matrix $\mathbf{R}$. It states that for every set of n points and for any integer n, k there exists a Lipchitz function

$f : \mathbb{R}^d \to \mathbb{R}^n$ such that for all points x, y the following holds:

$$(1-\epsilon)||x-y||^2 \leq ||f(x)-f(y)||^2 \leq (1+\epsilon)||x+y||^2. \tag{4.9}$$

Dimensionality reductions may be a fundamental step in medical image registration. Medical images can be characterized as points living in a high dimensional space. For example, a 512×512×512 MR image is a point in $2^{27} = 134217728$ dimensional space. This makes either learning or testing algorithms computationally expensive. Once we project the data in a lower dimensional space the JL lemma ensures that distances between the data will be approximately preserved. Therefore, it should be possible to perform image registration by evaluating the similarity measure in a much lower dimensional space, significanty speeding up image registration. Similar work has been proposed by [7] where they developed an image registration algorithm based on random projections of manifolds.

4.3 Summary

This thesis has explored machine learning techniques for learning an optimal similarity measure for use in medical image registration. In chapter 1 we described the general image registration problem and presented some practical applications. We presented in detail the three main components of image registration: the geometric transformation, the similarity measure and the optimization process. Finally, we briefly described our approach and presented related work. In chapter 2 we introduced the idea of machine learning, supervised and unsupervised learning associated with examples and stated the idea of distance metric learning (DML). We explained useful mathematical concepts and we described two approaches for DML. Finally we showed some experimental results for low dimensional data using those two approaches. In chapter 3 we described how we can learn an optimal similarity measure using DML and we applied this similarity measure

in the problem of brain CT/MR registration. We investigated the sensitivity of the learned similarity measure using ITML and finally we presented the experimental results to compare both learning techniques. In chapter 4 we presented conclusions and ideas for furture work for improving the registration results.

Bibliography

[1] Nathan D. Cahill. *Constructing and Solving Variational Image Registration Problems*. PhD thesis, University of Oxford, 2009.

[2] Jason V. Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S. Dhillon. Information-theoretic metric learning. *Proceedings of the 24th International Conference on Machine Learning*, 2007.

[3] Thomas G. Dietterich. Machine learning. *Department of Computer Science, Oregon State University*.

[4] Martin Guest. Introduction to manifolds. *Tokyo Metropolitan University*, 2007.

[5] W. Johnson and J. Lindenstrauss. Extensions of lipschitz mapping into a hilbert space. *Proceedings of the Conference on Modern Analysis and Probability*, pages 189–206, 1984.

[6] J. B. Antoine Maintz Josien P. W. Pluim and Max A. Viergever. Mutual information based registration of medical images: a survey. *IEEE Transactions on Medical Imaging*, XX(Y):31–45, 2003.

[7] Dennis M. Healy Jr. and Gustavo K. Rohde. Fast global image registration using random projections. In *IEEE International Symposium on Biomedical Imaging*, pages 476–479, 2007.

[8] Daewon Lee, Matthias Hofmann, Florian Steinke, Yasemin Altun, Nathan D. Cahill, and Bernhard SchG'Alkopf. Learning similarity measure for multi-modal 3d image registration. In *Computer Vision and Pattern Recognition*, pages 186–193, 2009.

[9] J. B. Antoine Maintz and Max A. Viergever. A survey of medical image registration. *Medical Image Analysis*, 2, 1998.

[10] J. Michael Fitzpatrick Principal Investigator. Retrospective image registration evaluation. *National Institutes of Health, Project Number 8R01EB002124-03*, Vanderbilt University, Nashville, TN.

[11] Hiroshi Nagaoka Shun-ichi Amari. *Methods of Information Geometry*. Oxford University Press, 2000.

[12] Jerome Friedman Trevor Hastie, Robert Tibshirani. *The Elements of Statistical Learning*. Springer, 2008.

[13] G. Tsagkatakis and A. Savakis. A random projections model for object tracking under variable pose and multi-camera views. *IEEE/ACM International Conference on Distributed Smart Cameras (ICDSC)*, 2009.

[14] Shijun Wang and Rong Jin. An information geometry approach for distance metric learning. *Proceedings of the 12th International Conference on Artificial Intelligence and Statistics*, 5, 2009.

[15] Kilian Q. Weinberger, John Blitzer, and Lawrence K. Saul. Distance metric learning for large margin nearest neighbor classification. In *In NIPS*. MIT Press, 2006.